METHOD FOR SECURE ENCODING OF DATA

FIELD OF INVENTION

[0001] The present invention relates to the secure encoding and electronic transmission of data by encoding sets of related data items into compact text strings. The present invention has particular but not exclusive application for financial transactions, product activation and registration and other types of program-to-program communications.

BACKGROUND

[0002] A number of methods and technologies have been developed to handle the transmission of data depending on the intended use of the data. For example, XML (Extensible Markup Language) allows the sender to construct common information formats and share both the format and the data on the World Wide Web, intranets, and elsewhere. Computer makers may agree on a standard or common way to describe the information about a computer product (processor speed, memory size, and so forth) and then describe the product information format with XML. Such a standard way of describing data would enable a user to send an intelligent agent (a program) to each computer maker's Web site, gather data, and then make a valid comparison. Another example is ASN.1 (Abstract Syntax Notation One), which is a standard way to describe a message (unit of application data) that can be sent or received in a network. ASN.1 has two parts: (1) the rules of syntax for describing the contents of a message in terms of data type and content sequence or structure and (2) how you actually encode each data item in a message. IDL (interface definition language) is a further example, and is a generic term for a language that lets a program or object written in one language communicate with another program written in an unknown language so that new objects can be sent to any platform environment and discover how to run in that environment.

[0003] There does not appear to be any methods or technologies that provide secure transmission of sets of related data items, encode the data set in a textual and compact manner

1

with an embedded integrity check and use a single key or template to determine the order and representation of the data.

OBJECT OF THE INVENTION

[0004] It is an object of the present invention to provide a method of encoding data that overcomes one or more of the aforementioned problems.

SUMMARY OF THE INVENTION

[0005] In one aspect the present invention broadly resides in a method for secure encoding of data including

constructing a template agreed for use by sender and receiver;

encoding data with reference to the template; and

decoding the data with reference to the template, wherein said template includes one or more element references, said element reference includes encoding attributes which determine the data element to encode or decode and the representation of each data element; said element references are arranged relative to each other in a format defined by the template.

[0006] In another aspect the invention broadly resides in a method for secure encoding of data including

constructing a template agreed to for use by sender and receiver;

encoding data with reference to the template; and

decoding the data with reference to the template, wherein said template includes one or more element references, said element reference includes encoding elements which describes the encoding of the data and data elements which represents the encoded data; said element references, encoding elements and data elements are arranged relative to each other in a format defined by the template.

2

[0007] In one preferred embodiment one or more data elements are spaced apart by one or more literal elements.

[0008] Where the data is to be transmitted, it is preferably transmitted after being encoded with reference to the template.

[0009] In another aspect the present invention broadly resides in a method for secure transmission of data including

constructing a template agreed for use by sender and receiver;

encoding data with reference to the template;

transmitting the encoded data; and

decoding the data with reference to the template, wherein said template includes one or more element references, said element reference includes encoding attributes which determine the data element to encode or decode and the representation of each data element; said element references are arranged relative to each other in a format defined by the template.

[0010] In another aspect the invention broadly resides in a method for secure transmission of data including

constructing a template agreed to for use by sender and receiver;

encoding data with reference to the template;

transmitting the encoded data; and

decoding the data with reference to the template, wherein said template includes one or more element references, said element reference includes encoding elements which describes the encoding of the data and data elements which represents the encoded data; said element references, encoding elements and data elements are arranged relative to each other in a format defined by the template.

[0011] Said element references are preferably spaced apart by one or more literal elements.

3

[0012] Preferably each template will vary in the way in which said element references and literal components are arranged. In one preferred embodiment each template varies in the representation used for each data element.

[0013] The template in use will depend on the context in which the dataset is used. For example, if the context concerns financial transactions, one of the data elements may be an account number. In contrast, a template used in the context of product activation may include a serial number as a data element.

[0014] The template serves to encode and decode the data and determine the order and representation of the individual elements within the template.

[0015] Literal components may occur before, between and or/after element references. Literal components are preferably copied literally in the encoding process and are verified character for character during decoding. In the template definition, literal components and element references are preferably distinguished by an established convention such as the use of letters, numbers and a limited set of punctuation characters for literal components and everything outside this set as lead-ins for element references. The representation of the template is preferably independent of the representation of the encoded data set.

[0016] Element references preferably specify the encoding details such as character, class, length, encryption flags; while the data element constitutes the data to encode or to recover when decoding, which may be represented as integers or multiples thereof and converted to strings in a variety of bases such as base 10 (decimal), base 16 (hexadecimal), base 24, base 36, base 57, base 64 or any other base suitably represented in a single character. Fixed length strings can be re-encoded into equivalent forms of multiple integers, variable length strings can be compressed and/or re-encoded thereby providing another level of protection and compactness in the encoded representation. The encoded representation is reduced in size by the use of large numerical bases in encoding the data elements.

[0017] The template may specify the application of any one of various types of integrity check, which provides a further layer of protection to the data. Preferably any element or combination of elements or another value mutually agreed to by the sender and recipient of the transmission can be used as a mutation seed that permutes selected elements and their encoded form as defined by the template. Mutation methods include, but are not limited to, binary and/or arithmetic operations involving the data element's original input value and one or more mutator values to produce the data element's output value.

BRIEF DESCRIPTION OF THE DRAWINGS

[0018] In order that the present invention be more readily understood and put into practical effect, reference will now be made to the accompanying drawings wherein:

Figures 1 to 4 are tables that set out an example syntax supporting six character classes;

Figure 5 is a table that sets out an example data set involved in a credit card transaction; and

Figure 6 is a table that sets out an example data set usable in an itemised shopping list.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

[0019] Due to the large variety of datasets that may be encoded using this method, template definitions can be tailored for each application. The parties that wish to exchange encoded datasets negotiate the characters, which will represent each individual element within the dataset to formulate an individual template. The choice of particular data elements, the arrangement of the elements and the type of encoding of the data provides a high level of encryption against external observation.

[0020] Individual elements (for example, account name, credit card number etc) are selected for a template for a particular dataset (of the data to be encoded). Where the data is lengthy or needs protection (such as a credit card number), the data is broken up into multiple elements thereby enhancing the encoding of that element.

5

[0021] Consideration is also given to the case (upper or lowercase), length of each element, and whether separators are to be used. Where separators are not used, encoded datasets will consist of a single uninterrupted string.

[0022] In the example syntax, an element reference specifies three, optionally four, aspects of the encoding of an element:

- the set of characters used,
- the number of characters,
- a symbolic reference to the element
- and optionally, additional attributes that influence the representation.

[0023] The character set used in encoding an element is expressed as one of the symbols given in Figure 1.

[0024] The number of characters with which a data element is encoded is specified as given in Figure 2. The values 0-9 correspond to numbers 1-10 respectively. That is, by way of example, if the length is given as 9 then the data element will be encoded in 10 characters.

[0025] The data element to be encoded or decoded is identified by a single character as given in Figure 3.

[0026] The optional encoding attributes supported by the example syntax are as given in Figure 4.

Example 1   *Credit Card Transaction*

[0027] Figure 5 shows the set of data elements that may be present within an online credit card transaction. The dataset includes card type, card number, card expiry date, currency, amount, cardholder name and checksum.

[0028] For the purposes of this example, the element t is used for the card type, elements n, u, m for three portions of the card number (up to eight digits each), element e for the expiry date, element c for the currency, element a for the amount and element C represents the checksum.

[0029] Using the template
#26%0:t-#10%7:n-#10%7:u-#10%7:m-#10%3:e-#26%2:c-#10%9:a-#16%7:C-
the dataset will be encoded as
Z-12345678-90123456-78901234-9909-AUD-1234567890-25A9DF50-
SAM SAMPLE

[0030] Each data element is encoded in accordance with its element reference. For example, in the above-mentioned credit card transaction, the template instructs that the card number is to be represented in three groups of 8 decimal digits each, separated by the literal component. This first example template produces a text string that preserves all element values in an easily human-readable representation.

[0031] Alternatively, using the template
#64%4:m$s#34%0:t$s#36%6:C$c#64%4:n$s#16%3:c$s#24%2:e$s#26%6:a$s#64%4
:u$s#64:.
the dataset will be encoded as
BBankL0AG7L8WYpzlB0E9F97XPLSEAXTEZ1AjoC+UMwrHe

[0032] This second template changes the order of the various data elements, mutates the data elements' original values, represents the mutated values in more compact character sets and as a result produces a more compact overall representation that obscures the transmitted data in a non-trivial fashion.

Example 2   _Itemized Shopping List_

[0033] Figure 6 shows a dataset that may be present on an itemised shopping list. For the purposes of this example, the element n is used for the item number, element c for the item count, element S for the SKU/ID, element u for the unit price and element C represents the checksum.

[0034] Using the template

#36%6:C$c#34%1:n$s#16%7:S$s#16%7:u$s#24%2:e$s#64:.

the dataset will be encoded as

0CEY196O751E8F424DE343895DKQWVlH8U/1fnSA9

[0035] This example template represents the dataset in a highly compact and encrypted form suitable for storage and/or transmission.

[0036] On the other hand, using the template

ItemNo   #10%2:n\n

Count   #10%2:c\n

SKU   #10%9:S\n

UnitPrice #10%9:u\n

Description :.

the dataset will be encoded as

ItemNo   001

Count   500

SKU   0000004567

UnitPrice 0001234567

Description 125 gram Yummy

[0037] In this example the template specifies literal components and encoding attributes aimed at representing the dataset in a tabular and readable form suitable for printing.

ADVANTAGES

[0038] An advantage of the present method of securely encoding and transmitting data using a template to produce an encoded text string is that the encoding of the data and the arrangements of the data elements provide a two-fold level of protection and can be decoded in a useable format with the template. Furthermore there is a third level of protection with the incorporation of a check sum in the dataset. The method allows data exchange between two computer programs and because of the highly compact nature of the encoded dataset, it can be transmitted quickly and efficiently.

VARIATIONS

[0039] It will of course be realised that while the foregoing has been given by way of illustrative example of this invention, all such and other modifications and variations thereto as would be apparent to persons skilled in the art are deemed to fall within the broad scope and ambit of this invention as is herein set forth.

[0040] Throughout the description and claims this specification the word "comprise" and variations of that word such as "comprises" and "comprising", are not intended to exclude other additives, components, integers or steps.